REMARKS

This is in response to the Office Action mailed 11/20/2007. Reconsideration of this application is respectfully requested in view of this response/amendment.

STATUS OF CLAIMS

Claims 30-44 are pending.

Claims 33 and 43 are objected to because of informalities due to minor grammatical errors.

Claims 30, 31, 36, 40 and 41 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicant regards as the invention.

Claims 30-44 stand rejected under 35 U.S.C. § 102(a) as being anticipated by the article to Franz entitled, "An Efficient XML Schema Typing System," published 11/18/2003, previously cited.

OVERVIEW OF CLAIMED INVENTION

The present invention, in one embodiment, provides for a computer-based method for validating a fragment of a structured document, wherein the computer-based method comprises steps of: (a) receiving as input a fragment of an XML document into a runtime validation engine; and (b) outputting a validation pass message as follows: (i) obtaining a first token from said fragment of said XML document, (ii) determining whether said first token is of element type said fragment of said XML document that is to be validated against, and if so, (iii) obtaining next token from said fragment of said XML document, (iv) checking whether said next token signifies end of said fragment of said XML document, and if so, returning a validation pass if an annotated automaton encoding (AAE) stack is empty; and, if said next token does not signify end of said fragment of said XML document, continuing validation as in validating an entire XML document, and when successfully validated as in an entire XML document, returning to step (iii) until end of said fragment of said XML document token is received, wherein, when first token is not of said element type, or when said continued validation as in validating an entire document fails in step iv or when said AAE stack is not empty, said method returns a validation failure message.

CLAIM OBJECTIONS

Claims 33 and 43 are objected to because of minor grammatical errors. Applicants are appreciative of the Examiner for pointing out the grammatical errors in claims 33 and 43. Appropriate correction has been made via the current amendment without adding new matter.

REJECTIONS UNDER 35 U.S.C. § 112

Claims 30, 31, 36, 40 and 41 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicant regards as the invention. As per the Examiner's recommendations the bullet points have been removed. Further, ambiguities regarding the phraseology of "step iii" and "step iv" have been removed via a minor amendment. Such amendments have been made for clarification purposes only and the amendments have been made without adding new matter.

Applicants wish to note that the deletions of the bullet points in the claims are not clear, so Applicants have also provided a clean copy of the claims in **Appendix A**, which accompanies the current amendment.

REJECTIONS UNDER 35 U.S.C. § 102(a)

Claims 30-44 stand rejected under 35 U.S.C. § 102(a) as being anticipated by the article to Franz et al. entitled, "An Efficient XML Schema Typing System," published 11/18/2003, previously cited. The Manual for Patenting Examining Procedure (MPEP) §2131 clearly sets forth the standard for rejecting a claim under 35 U.S.C. §102(b). "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." (MPEP §2131, quoting Verdegaal Bros. v. Union Oil Co. of California 2 USPQ2d 1051, 1053 (Fed Cir. 1987)). Applicants respectfully assert, and as will be shown in the arguments below, that Franz et al. fails to teach the claimed invention as required by the MPEP.

With regard to the feature in independent claim 30 of "determining whether said first token is of element type said fragment of said structured document that is to be validated against," the Examiner on page 4 of the Office Action cites line 7 of Franz's pseudo code shown on page 12 (and accompanying description in pages 10-11 of Franz) as teaching such a feature. Line 7 of the Franz's pseudo code is reproduced verbatim below:

"**If** (*FSA*($\tau_{c:x}$) *accepts token*)"

By Franz's own admission on page 10 (last sentence), *FSA*($\tau_{c:x}$) "**denotes the Finite State Automaton of complex type $\tau_{c:x}$**" and **makes NO mention or suggestion for making a**

determination as to whether a first token is of element type corresponding to the fragment. Applicants respectfully submit to the Examiner that **a mere mention of whether or not a Finite State Automaton accepts a token CANNOT be equated to the Applicants' feature of determining whether a first token is of element type corresponding to the fragment.** Absent such a teaching, Franz et al. CANNOT anticipate or render obvious the teachings of independent claim 30.

With regard to the feature in independent claim 30 of "obtaining next token from said fragment of said structured document," the Examiner on page 4 of the Office Action cites line 9-10 of Franz's pseudo code shown on page 12 as teaching such a feature. Lines 9-10 of Franz's pseudo code are reproduced verbatim below:

"$\tau_y \leftarrow typing(\tau_{c:x}, a_i)$

a_i *is annotated with* τ_y"

The above-mentioned pseudo code, by Franz et al.'s own admission on page 10 (in §3.3) merely references a "**typing function**" that "**provides sufficient information for type annotations**". The Examiner's citation in the pseudo code, or the corresponding description in the Franz et al. reference, makes NO mention or suggestion for the step of obtaining the next token from a fragment of a structured document. Applicants respectfully submit that a **mere mention of a "typing function" CANNOT be equated to the Applicants' feature of obtaining the next token from a fragment of a structured document**. Absent such a teaching, Franz et al. CANNOT anticipate or render obvious the teachings of independent claim 30.

With regard to the feature in independent claim 30 of "checking whether said next token signifies end of said fragment of said structured document, and if so, returning a validation pass if an annotated automaton encoding (AAE) stack is empty," the Examiner on page 4 of the Office Action cites lines 24-25 of Franz's pseudo code shown on page 12 as teaching such a feature. Lines 24-25 of Franz's pseudo code are reproduced verbatim below:

"**if** (*FSA*($\tau_{c:x}$) reaches an accept state)

Then *pop(stack)*, $\tau_{c:x} \leftarrow$ *peek*(*stack*) **fi**"

The above-mentioned pseudo code makes NO mention or suggestion for **checking if the next token signifies the end of a fragment** and it makes NO mention or suggestion of **returning a validation pass if the AAE stack is empty**. Absent such teachings, Franz et al. CANNOT anticipate or render obvious the teachings of independent claim 30.

Further, with regard to Applicants feature of "if said next token does not signify end of said fragment of said structured document, continuing validation as in validating an entire structured document, and when successfully validated as in an entire structured document, returning to step (iii) until end of said structured document token is received and outputting a validation pass when AAE stack is empty," the Examiner states in the response of 11/20/2007 that such a feature is generally disclosed in lines 4-26 of the pseudo code shown on page 12 of Franz et al. However, a closer reading of the pseudo, in conjunction with the arguments presented above, merely suggests that a type validation algorithm and makes no mention of the specific steps of "continuing validation as in validating an entire structured document" and "returning to step (iii) until end of said structured document token is received and outputting a validation pass when AAE stack is empty." Absent such teachings, Franz et al. CANNOT anticipate or render obvious the teachings of independent claim 30.

Hence at least for the reasons set forth above, Applicants believe that independent claim 30 reads over the art of record and is in allowable form. Therefore, Applicants respectfully request the Examiner to withdraw the 35 U.S.C. §102 rejection with respect to independent claim 30.

If the examiner still feels that that the features of independent claim 30 are disclosed in the Franz reference, Applicants respectfully remind the examiner that it is the duty of the examiner to **specifically point out** each and every limitation of a claim being rejected as per §1.104(c)(2) of Title 37 of the Code of Federal Regulations and section 707 of the M.P.E.P., which explicitly states that "the particular part relied on must be designated" and "the pertinence of each reference, if not apparent, must be clearly explained and each rejected claim specified".

The Examiner has provided the same reasoning, as above, for independent claims 36 and 40. Therefore, the above-mentioned arguments with respect independent claim 30 substantially apply to independent claims 36 and 40. Hence at least for the reasons set forth above, Applicants believe that independent claims 36 and 40 also read over the art of record and are in allowable form. Therefore, Applicants respectfully request the Examiner to withdraw the 35 U.S.C. §102 rejection with respect to independent claims 36 and 40.

The above-mentioned arguments for independent claims 30, 36, and 40 substantially apply to dependent claims 31-35, 37-39, and 41-44 as they depend from an allowable claim.

SUMMARY

As has been detailed above, none of the references, cited or applied, provide for the specific claimed details of Applicants' presently claimed invention, nor renders them obvious. It is believed that this case is in condition for allowance and reconsideration thereof and early issuance is respectfully requested.

This response is being timely filed and therefore no request or fee for an extension of time has been included. However, the Commissioner is hereby authorized to charge any deficiencies in the fees provided to Deposit Account No. 09-0460.

If it is felt that an interview would expedite prosecution of this application, please do not hesitate to contact Applicants' representative at the below number.

Respectfully submitted,

/ramraj soundararajan/

Ramraj Soundararajan
Registration No. 53,832

IP Authority, LLC.
4821A Eisenhower Ave
Alexandria, VA 22304
(703) 461-7060

February 20, 2008

APPENDIX A

(CLEAN COPY OF CLAIMS)

30. (Currently Amended) A computer-based method for validating a fragment of a structured document, said computer-based method implemented in computer readable program code, said computer readable program code stored in computer memory, said computer-based method comprising steps of:

(a) receiving as input a fragment of a structured document into a runtime validation engine;

(b) outputting a validation pass message as follows:

(i) obtaining a first token from said fragment of said structured document,

(ii) determining whether said first token is of element type said fragment of said structured document that is to be validated against, and if so,

(iii) obtaining next token from said fragment of said structured document,

(iv) checking whether said next token signifies end of said fragment of said structured document, and if so, returning a validation pass if an annotated automaton encoding (AAE) stack is empty; and

if said next token does not signify end of said fragment of said structured document, continuing validation as in validating an entire structured document, and when successfully validated as in an entire structured document, returning to step (iii) until end of said structured document token is received and outputting a validation pass when AAE stack is empty.

31. (Currently Amended) The computer-based method of claim 30, wherein, when first token is not of said element type, or when said continued validation as in validating an entire document fails in step (iv) or when said AAE stack is not empty, said method returns a validation failure message.

32. (original) The computer-based method of claim 30, wherein said structured document is an XML document.

33. (Currently Amended) The computer-based method of claim 30, wherein said first or next token is either an element type name or an attribute name.

34. (original) The computer-based method of claim 30, wherein said first or next token is a lexeme, said lexeme being any of the following: a start tag name, an attribute name, or an end tag name.

35. (original) The computer-based method of claim 30, wherein said computer-based method is implemented in conjunction with a database.

36. (Currently Amended) A computer-based method for validating a fragment of a structured document, said computer-based method implemented in computer readable program code, said computer readable program code stored in computer memory, said computer-based method comprising steps of:

(a) receiving as input a fragment of an XML document into a runtime validation engine;

(b) outputting a validation pass message as follows:

(i) obtaining a first token from said fragment of said XML document,

(ii) determining whether said first token is of element type said fragment of said XML document that is to be validated against, and if so,

(iii) obtaining next token from said fragment of said XML document,

(iv) checking whether said next token signifies end of said fragment of said XML document, and if so, returning a validation pass if an annotated automaton encoding (AAE) stack is empty; and

if said next token does not signify end of said fragment of said XML document, continuing validation as in validating an entire XML document, and when successfully validated as in an entire XML document, returning to step (iii) until end of said fragment of said XML document token is received, wherein, when first token is not of said element type, or when said continued validation as in validating an entire document fails in step(iv) or when said AAE stack is not empty, said method returns a validation failure message.

37. (original) The computer-based method of claim 36, wherein said first or next token is either an element type name or an attribute name.

38. (original) The computer-based method of claim 36, wherein said first or next token is a lexeme, said lexeme being any of the following: a start tag name, an attribute name, or an end tag name.

39. (original) The computer-based method of claim 36, wherein said computer-based method is implemented in conjunction with a database.

40. (Currently Amended) An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements a computer-based method for validating a fragment of a structured document, said computer-based method implemented in computer readable program code, said computer readable program code stored in computer memory, said computer usable medium comprising:

(a) computer readable program code aiding in receiving as input a fragment of a structured document into a runtime validation engine;

(b) computer readable program code aiding in outputting a validation pass message as follows:

(i) computer readable program code aiding in obtaining a first token from said fragment of said structured document,

(ii) computer readable program code determining whether said first token is of element type said fragment of said structured document that is to be validated against, and if so,

(iii) computer readable program code aiding in obtaining next token from said fragment of said structured document,

(iv) computer readable program code checking whether said next token signifies end of said fragment of said structured document, and if so, returning a validation pass if an annotated automaton encoding (AAE) stack is empty; and

if said next token does not signify end of said fragment of said structured document, computer readable program code continuing validation as in validating an entire structured document, and when successfully validated as in an entire structured document, computer readable program code returning to step (iii) until end of said structured document token is received and outputting a validation pass when AAE stack is empty.

41. (Currently Amended) The article of manufacture of claim 40, wherein, when first token is not of said element type, or when said continued validation as in validating an entire document fails in step (iv) or when said AAE stack is not empty, computer readable program code returns a validation failure message.

42. (**original**) The article of manufacture of claim 40, wherein said structured document is an XML document.

43. (Currently Amended) The article of manufacture of claim 40, wherein said first or next token is either an element type name or an attribute name.

44. (**original**) The article of manufacture of claim 40, wherein said first or next token is a lexeme, said lexeme being any of the following: a start tag name, an attribute name, or an end tag name.